



**Akademia Nauk Stosowanych**  
**im. Hipolita Cegielskiego w Gnieźnie Uczelnia Państwowa**

**SYLABUS**

<b>Pozycja przedmiotu w planie:</b>		R.II/S.IV - 4
<b>1. OGÓLNY OPIS PRZEDMIOTU</b>		
1	Nazwa modułu	Moduł zajęć kierunkowych
2	Nazwa przedmiotu	Inżynieria oprogramowania I
3	Kierunek studiów	Informatyka
4	Poziom studiów	studia pierwszego stopnia
5	Forma studiów	niestacjonarne
6	Profil studiów	praktyczny
7	Rok studiów	drugi
8	Semestr przedmiotu	czwarty
9	Jednostka prowadząca kierunek studiów	Instytut Nauk Technicznych
10	Liczba punktów ECTS	3
11	Sposób zaliczenia:	Wykład: zaliczenie z oceną Projekt: zaliczenie z oceną
12	Imię i nazwisko nauczyciela (li) akademickiego (ich), stopień lub tytuł naukowy, adres e-mail	dr inż. Piotr Sujka p.sujka@ans-gniezno.edu.pl
13	Imię i nazwisko koordynatora(ów) przedmiotu, stopień lub tytuł naukowy, adres e-mail	dr inż. Piotr Sujka p.sujka@ans-gniezno.edu.pl
14	Język wykładowy	polski
15	Tryb prowadzenia zajęć	W salach i zdalny
16	Sposób prowadzenia zajęć	Synchroniczny i asynchroniczny
17	Narzędzia informatyczne wykorzystywane do prowadzenia zajęć, udostępniania materiałów i komunikacji ze studentami	Platforma Moodle. MS Teams
18	Przedmioty wprowadzające	
19	Wymagania wstępne	1. Podstawowe wiadomości z zakresu technik programowania 2. Umiejętność efektywnego samokształcenia w dziedzinach związanych z informatyką jako wybranym kierunkiem studiów 3. Student ma świadomość konieczności poszerzania swoich kompetencji oraz gotowość do podjęcia współpracy w ramach zespołu.
20	<b>Cele przedmiotu:</b>	
C1	Poznanie zasad i metod wytwarzania oprogramowania w sposób systematyczny.	
C2	Student przyswaja sobie umiejętności: posługiwania się wzorcami projektowymi; projektowania oprogramowania zgodnie z metodyką strukturalną.	

<b>C3</b>	Dokonywanie przeglądu projektu oprogramowania.	
<b>21</b>	Forma zajęć, liczba godzin wymagająca bezpośredniego udziału nauczyciela akademickiego, liczba godzin nakładu pracy studenta	
	<b>Forma zajęć</b>	<b>Liczba godzin</b>
	1. Wykład	16
	2. Projekt	16
	<b>Suma godzin</b>	<b>32</b>
<b>lp.</b>	<b>Całkowity nakład pracy studenta</b>	
	Nakład pracy związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela akademickiego wynosi:	<b>Godzinowe obciążenie studenta</b>
<b>1.</b>	Udział w wykładach 16 godzin Udział w projektach 16 godzin	32 godziny
	Nakład pracy związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela akademickiego wynosi 32 godziny, co odpowiada 1 punktowi ECTS.	
<b>2</b>	Bilans nakładu pracy studenta: Samodzielne studiowanie literatury: 20 godzin Przygotowanie do zaliczenia z: 8 godzin Wykonywanie projektów: 20 godzin  Łączny nakład pracy studenta wynosi 48 godzin, co odpowiada 2 punktom ECTS.	48 godzin
<b>3</b>	<b>Łączny nakład pracy studenta (pozycja 1+2)</b>	80 godzin
<b>4</b>	<b>Punkty ECTS za przedmiot</b>	3 ECTS
<b>5</b>	Liczba punktów ECTS, którą student musi osiągnąć w ramach zajęć o charakterze praktycznym w tym zajęć laboratoryjnych, warsztatowych, projektowych	1,5 ECTS
Efekty uczenia się - wiedza	<p>K_W17: Student ma wiedzę w zakresie projektowania oprogramowania, narzędzi i środowisk wytwarzania oprogramowania, procesów wytwarzania oprogramowania; ma wiedzę w zakresie specyfikacji wymagań, walidacji i testowania oprogramowania, zna metody zarządzania przedsięwzięciami programistycznymi oraz ich jakością.</p> <p>K_W18: Student ma wiedzę o cyklu życia oprogramowania, oraz cyklu życia układów cyfrowych, sprzętu komputerowego, sieciowego i systemów wbudowanych, ma wiedzę o stanie oraz najnowszych trendach i uwarunkowaniach rozwojowych informatyki, elektroniki, automatyki, robotyki i systemów ICT (IoT, BIoT, IIoT).</p> <p>K_W19: Student ma podstawową wiedzę w zakresie budowy relacyjnych i nierelacyjnych systemów baz danych, modelowania danych, projektowania relacyjnych i nierelacyjnych bazy danych, języków zapytań do baz danych, definicji danych oraz przetwarzania transakcji.</p>	
Efekty uczenia się - umiejętności	<p>K_U07: Student potrafi utworzyć specyfikację, zaprojektować i zaimplementować system informatyczny, cyfrowy lub wbudowany z zastosowaniem wybranych narzędzi wspierających budowę oprogramowania, wzorców projektowych, zgodnie z opracowanym harmonogramem.</p> <p>K_U08: Student ma umiejętności: posługiwania się wzorcami projektowymi; projektowania oprogramowania zgodnie z metodyką strukturalną; dokonywania przeglądu projektu oprogramowania; wybierania narzędzi wspomagających budowę oprogramowania; doboru modelu procesu wytwarzania oprogramowania do specyfiki przedsięwzięcia; specyfikowania wymagań dotyczących oprogramowania i przeprowadzania ich przeglądu; tworzenia, oceny i realizacji planu testowania; uczestniczenia w inspekcji kodu; zarządzania konfiguracją oprogramowania; opracowywania planu przedsięwzięcia dotyczącego budowy oprogramowania.</p>	
Efekty uczenia się – kompetencje społeczne	K_K01: Student rozumie potrzebę i zna możliwości ciągłego dokształcania się, krytycznie odnosi się do posiadanej wiedzy, podnoszenia kompetencji zawodowych, osobistych i	

	<p>społecznych.</p> <p>K_K02: Student ma świadomość ważności i rozumie pozatechniczne aspekty i skutki działalności inżyniera-informatyka, w tym jej wpływ na środowisko i związaną z tym odpowiedzialność za podejmowane decyzje; dba o dobre tradycje zawodu informatyka.</p> <p>K_K04: Student ma świadomość odpowiedzialności za pracę własną oraz gotowość podporządkowania się zasadom pracy w zespole i ponoszenia odpowiedzialności za wspólnie realizowane zadania, potrafi określić priorytety służące realizacji określonego przez siebie lub zespół zadania.</p>
--	--

2. TREŚCI PROGRAMOWE ODNIESIONE DO EFEKTÓW UCZENIA SIĘ		
Forma zajęć	Treści programowe	liczba godzin
<b>Forma: Wykład</b>		
W1	Przedmiot zainteresowania inżynierii oprogramowania.	2
W2	Narzędzia i środowiska programistyczne.	2
W3	Procesy wytwarzania oprogramowania.	2
W4	Zbieranie oraz zarządzanie wymaganiami, specyfikacja wymagań.	2
W5	Analiza systemów informatycznych.	2
W6	Zarządzanie konfiguracją i wersjonowanie.	2
W7	Pielęgnacja i refaktoryzacja.	2
W8	Procesy wytwarzania oprogramowania - dokumentowanie procesu i produktu programowego.	2
<b>Forma: Projekt</b>		
P1	Specyfikacja wymagań.	2
P2	Opis słowny działania programu.	1
P3	Identyfikacja obiektów danych	1
P4	Diagram encja-związek	2
P5	Diagram przepływu danych	2
P6	Projekt architektury systemu	2
P7	Projekt interfejsu użytkownika	2
P8	Projekt procedury	2
P9	Testowanie ścieżek podstawowych	2

3. Literatura	
Literatura podstawowa	<ol style="list-style-type: none"> <li>Sommerville I., <i>Inżynieria oprogramowania</i>. WNT, Warszawa 2005.</li> <li>Sacha, <i>Inżynieria oprogramowania</i>, PWN, 2010.</li> </ol>
Literatura uzupełniająca	<ol style="list-style-type: none"> <li>Pressman R., <i>Praktyczne podejście do inżynierii oprogramowania</i>. WNT, Warszawa 2004.</li> <li>Jaskiewicz A., <i>Inżynieria oprogramowania</i>. Helion, Gliwice 1997</li> </ol>

4. Metody dydaktyczne	
<b>Forma</b>	<b>Metody dydaktyczne</b>
<b>Wykład</b>	Wykład informacyjny (prezentacje multimedialne, materiały na platformie Moodle).
<b>Projekt</b>	Projekt, dyskusja.

5. Metody i kryteria oceniania															
Forma zajęć:	Forma zaliczenia:														
<p>Uzyskane punkty są przeliczane na oceny według następującej skali:</p> <table> <tr> <td>Procent punktów</td> <td>Ocena</td> </tr> <tr> <td>91-100%</td> <td>Bardzo dobry</td> </tr> <tr> <td>85-90%</td> <td>Dobry plus</td> </tr> <tr> <td>76-84%</td> <td>Dobry</td> </tr> <tr> <td>66-75%</td> <td>Dostateczny plus</td> </tr> <tr> <td>51-65%</td> <td>Dostateczny</td> </tr> <tr> <td>0-50%</td> <td>Niedostateczny</td> </tr> </table>		Procent punktów	Ocena	91-100%	Bardzo dobry	85-90%	Dobry plus	76-84%	Dobry	66-75%	Dostateczny plus	51-65%	Dostateczny	0-50%	Niedostateczny
Procent punktów	Ocena														
91-100%	Bardzo dobry														
85-90%	Dobry plus														
76-84%	Dobry														
66-75%	Dostateczny plus														
51-65%	Dostateczny														
0-50%	Niedostateczny														
<p>Opis:</p> <p><b>Forma zaliczenia: wykłady</b> Ocena końcowa na podstawie wykonanych aktywności (lekcje, testy) na platformie Moodle. Opcjonalnie - kolokwium zaliczeniowe z całości materiału.</p> <p><b>Forma zaliczenia: projekty</b> Zaliczenie końcowe na podstawie wykonania poszczególnych etapów projektu i zadań. Wykonanie poszczególnych zadań jest oceniane na bieżąco po zakończeniu ćwiczenia.</p>															
Warunkiem zaliczenia przedmiotu jest uzyskanie pozytywnych ocen z każdej z form zajęć															

	Zatwierdzenie karty opisu zajęć	Podpis
	Stanowisko Tytuł/stopień naukowy, imię nazwisko	
<b>Opracował</b>	dr inż. Piotr Sujka	
<b>Zatwierdził</b>	Dyrektor Instytutu Nauk Technicznych dr inż. Łukasz Józefowski	